



PATENT ABSTRACTS OF JAPAN

(11) Publication number: **11065920 A**(43) Date of publication of application: **09.03.99**

(51) Int. Cl. **G06F 12/04**
G11C 7/00
G11C 11/407

(21) Application number: **09216069**(71) Applicant: **NKK CORP**(22) Date of filing: **11.08.97**(72) Inventor: **NAKANO YASUO**

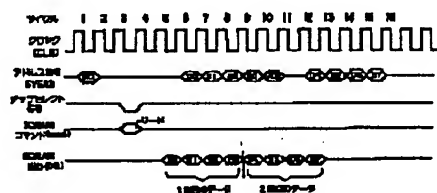
(54) MEMORY CONTROL METHOD

(57) Abstract:

PROBLEM TO BE SOLVED: To realize a high speed processing by providing a buffer for desired bits inside a memory controller, arbitrarily setting burst length, and reading the data of the second address following the first address at the time of reading the first address, and temporarily storing the data in the buffer.

SOLUTION: A buffer for 32 bytes is provided inside a memory controller by using a synchronous dynamic RAM (SDRAM), and burst length is set as 8, and the data of the next address are already first-read. That is, data D0-D3 and D4-D7 for two addresses, that is, an initial address AD1 and a next address AD2 are read in a cache area, and the data D4-D7 of the first-read next address AD2 are stored. Then, when the address of the block read request of a processor is made coincident with the address AD2 of the stored data, the read data are returned at a high speed.

COPYRIGHT: (C)1999,JPO



(19)日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11)特許出願公開番号

特開平11-65920

(43)公開日 平成11年(1999)3月9日

(51)Int.Cl. ^a	識別記号	F I
G 0 6 F 12/04	5 1 0	G 0 6 F 12/04 5 1 0 C
G 1 1 C 7/00	3 1 2	G 1 1 C 7/00 3 1 2 Z
11/407		11/34 3 6 2 S

審査請求 未請求 請求項の数3 O L (全 4 頁)

(21)出願番号 特願平9-216069

(22)出願日 平成9年(1997)8月11日

(71)出願人 000004123

日本鋼管株式会社

東京都千代田区丸の内一丁目1番2号

(72)発明者 中野 泰生

東京都千代田区丸の内一丁目1番2号 日

本鋼管株式会社内

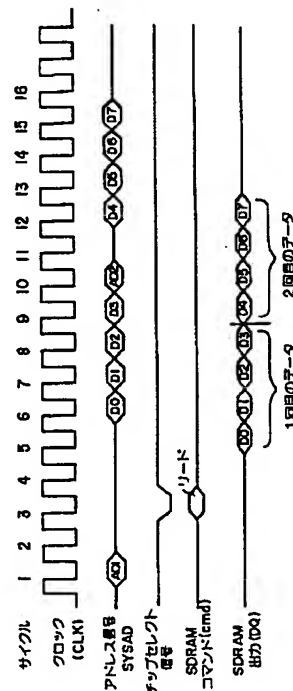
(74)復代理人 弁理士 花輪 義男

(54)【発明の名称】 メモリ制御方法

(57)【要約】

【課題】従来のDRAMに対するデータの読み出し／書き込み動作にかかる時間は、RASのサイクル時間により制限されているため、読み込みの回数が多いほど、回数分のRASのサイクル時間が加算され、高速化が難しかった。

【解決手段】本発明は、SynchronousDRAMを用いて、メモリコントローラ内にバッファを設け、SDRAMを64bitバス幅になるよう並列に並べて、バーストの長さを設定し、常に次のアドレスのデータを先読みし、高速処理を実現するメモリ制御方法である。



【特許請求の範囲】

【請求項1】 読み出し／書き込み動作を行うために指定されたアドレスを内部的にインクリメントする機能を有する半導体メモリ装置のメモリ制御方法において、メモリ制御部内部に所望するビット分のバッファを設け、バースト長を任意に設定して、最初の第1のアドレスに続く第2のアドレスのデータを前記第1のアドレスによるデータを読み出し時に併せて読み出し、前記バッファに一時的に蓄積し、前記第1のアドレスのデータ処理の終了後の新たに処理するアドレスが前記第2のアドレスと一致した場合に、前記バッファから前記第2のアドレスのデータを読み出し、データ処理することを特徴とするメモリ制御方法。

【請求項2】 前記第1のアドレスのデータのバースト長の2倍のバースト長を設定することにより、前記第1のアドレスに続く第2のアドレスのデータを前記第1のアドレスのデータ読み出し時に併せて読み出し前記バッファに一時的に蓄積することを特徴とする請求項1に記載のメモリ制御方法。

【請求項3】 前記メモリ制御方法を行う半導体メモリ装置に、Synchronous DRAMを用いて、メモリ制御を行うことを特徴とする請求項1に記載のメモリ制御方法。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】本発明は、ダイナミックRAMにおけるデータの読み出し／書き込み動作の高速化を図るメモリ制御方法に関する。

【0002】

【従来の技術】従来、ダイナミックRAM (DRAM) は、最も汎用される揮発性メモリとして電子計算機の主記憶装置のような大容量システムから、OA機器、パーソナルコンピュータ、ゲーム機器の小容量システムに至るまで広範囲に使用されている。このDRAMの読み出し動作や書き込み動作は、行アドレス入力によりワード線を選択して、そのワード線につながる全メモリセルをビット線群に接続し、更に列アドレス入力により特定のビット線を選択してデータ線と接続することによって、所望するメモリセルにランダムにアクセス可能なリード／ライト (read/write) モードが基本となる。この動作では、1回のメモリサイクル内にデータの読出し若しくは、書き込みの一方だけが行われ、サイクルの終了と共に周辺回路が初期化されて、次サイクルへの待機状態に入る。

【0003】しかし、動作時間の高速化が要望されるに伴って、DRAM自体にも、最適化が加えられている。

【0004】例えば、行列状に配列されたメモリセルの位置を指定するアドレス入力信号は行アドレス及び列アドレスの2種を入力する必要があるが、この2種のアドレス信号を同一の入力端子から時分割して入力する手法がある。

【0005】即ち、RAS (row address stobe) 及び CAS (column address strobe) と称される2種の外部クロック信号を生成し、まず、行アドレスをアドレス端子に印加した後、RASを入力すると、アドレス信号が内部に取り込まれ、行アドレスとしてラッチされる。次に、列アドレスを同一のアドレス端子に印加して、CASを入力すると、同様に列アドレスがラッチされる。

【0006】このような時分割したアドレス入力方法は、アドレスマルチプレクス (address multiplexing) と称されており、メモリ素子のピン数を減らしてパッケージを小形化し、実装密度の向上に大きく寄与している。

【0007】

【発明が解決しようとする課題】前述した時分割したアドレス入力方法において、プロセッサの1キャッシュ・ラインが32バイトであった場合には、64bit×4として構成される。

【0008】これに伴い、プロセッサはメモリセルからデータを読み出すとき、キャッシュ領域の場合には、複数のブロックリード (64bitのアクセスを4回) を行うこととなる。

【0009】これらのRAS及びCASの動作タイミングは、図2に示すように、活性化期間とプリチャージ期間を交互に繰り返すことが必要で、RASのサイクル時間がDRAMのサイクル時間となる。

【0010】RASがHレベルの期間は、RAS系内部回路のプリチャージ期間であり、メモリとしての動作は行われない。Lレベルの期間は、アドレスバッファからセンスアンプまでが活性化される活性化期間となり、行アドレスの入力、行デコードの活性化、ワード線の選択、センス動作等にかかる時間に、読み出し／書き込み動作に必要な時間が加算された期間である。また、CASがHレベルの期間は、CAS系内部回路のプリチャージ期間であり、この間は読出し／書き込み動作は行われない。CASがLレベルの期間は、その時点での列アドレスがラッチされ、これが所望の列アドレス入力として有効となる。その後CASがLレベルの期間は、新たな列アドレスを受けつけない。DRAMのサイクル時間は、RASのサイクル時間により制限されているため、読み込みの回数が多いほど、回数分のRASのサイクル時間が加算されることとなり、処理の高速化が難しくなる。

【0011】そこで本発明は、メモリコントローラ内にバッファを設け、常に次のアドレスのデータを先読みし、高速処理を実現するメモリ制御方法を提供することを目的とする。

【0012】

【課題を解決するための手段】本発明は上記目的を達成するために、メモリ制御部内部に所望するビット分のバッファを設け、バースト長を任意に設定して、最初の第

1のアドレスに続く第2のアドレスのデータを前記第1のアドレスを読み出し時に併せて読み出し、前記バッファに一時的に蓄積し、前記第1のアドレスのデータ処理の終了後の新たに処理するアドレスが前記第2のアドレスと一致した場合に、前記バッファから前記第2のアドレスのデータを読み出し、データ処理するメモリ制御方法を提供する。

【0013】以上のような構成のメモリ制御方法は、Synchronous DRAMを用いて、メモリコントローラ内にバッファを設け、SDRAMを64bitバス幅になるよう並列に並べて、バーストの長さを設定し、常に次のアドレスのデータを先読みし、高速処理を実現する。

【0014】

【発明の実施の形態】以下、図面を参照して本発明の実施形態について詳細に説明する。

【0015】図1には、本発明のメモリ制御方法を実現するためのメモリコントローラを設計した場合のタイミングチャートの例を示す。

【0016】前述した様に、プロセッサはメモリからデータを読み出す時、キャッシュ領域に、ブロックリード(64bitのアクセスを4回)を行っている。

【0017】前述したStandard DRAMやEDODRAMでブロックリードを行う場合は、あるRASをアクティブにし、その後CASを4回アクティブにして、所望のデータをメモリから読み込むことになる。つまり、4回の読み出しを行っている。これに対し、Synchronous DRAM(以下、SDRAM)の場合は、あるアドレスにリード・コマンドを与えることにより、SDRAM内部でアドレスのインクリメントが自動的に行われ、バースト動作が完了する(CASを4回アクティブにする必要がない)。

【0018】また、SDRAMではバーストの長さ(バースト長)についても、レジスタの設定により段階的に設定された値(例えば、1、2、4、8)の中から選択することができる。

【0019】通常、64bit CPUがSDRAMを用いてシステムを構成する場合、SDRAMを64bitバス幅になるよう並列に並べ、バースト長を4と設定してプロセッサのブロック・リードに対応することになる。

【0020】通常、システムのクロックを66MHzに設定した場合、9サイクルで32バイトのデータ転送が完了しているため、パフォーマンスは約237MBit/sec程度となる。

【0021】ところが、SDRAMにおいては、バースト長を8と設定することも可能である。本実施形態では、SDRAMでバースト長を8と設定することにより、あるアドレスへのブロック・リードを行った際、次のアドレスのデータも先読みすることが可能となる。

【0022】即ち、メモリコントローラ内部に32バイト分のバッファを確保し、初回のアドレスAD1と次回のアドレスAD2の2つ分のアドレスのデータD0~D3、D4~D7をキャッシュ領域に読み出して、先読みした次のアドレスAD2のデータD4~D7を蓄えておけば、次のプロセッサのブロック・リードリクエストのアドレスが蓄えていたデータのアドレスAD2と一致した場合、読み出したデータを高速に返送することができる。但し、次のアドレスが異なった場合には、先読みされたデータは無効となる。

【0023】連続したアドレスへのブロック・リードが継続した場合、1回目(初回)のブロック・リードでは、9サイクルを要しているが、2回目(次回)のアクセスでは、予め先読みして内部バッファに蓄えられたデータD4~D7を、そのまま返送すれば良いため、6サイクルしか必要としない。

【0024】つまり、連続するアドレスがアクセスされる場合、偶数回目は、常に6サイクルで良いことになる。従って、従来は、64バイトの転送は、18サイクルの期間が必要であったものが、本実施形態を利用すれば、15サイクル(つまり、9+6=15)で実現されるため、ハード的な改善をしなくとも、パフォーマンスは、約284MBit/secとなり、従来に比べて、20%程度のパフォーマンスが向上される。

【0025】

【発明の効果】以上詳述したように本発明によれば、メモリコントローラ内にバッファを設け、常に次のアドレスのデータを先読みし、高速処理を実現するメモリ制御方法を提供することができる。

【図面の簡単な説明】

【図1】本発明のメモリ制御方法を説明するためのタイミングチャートである。

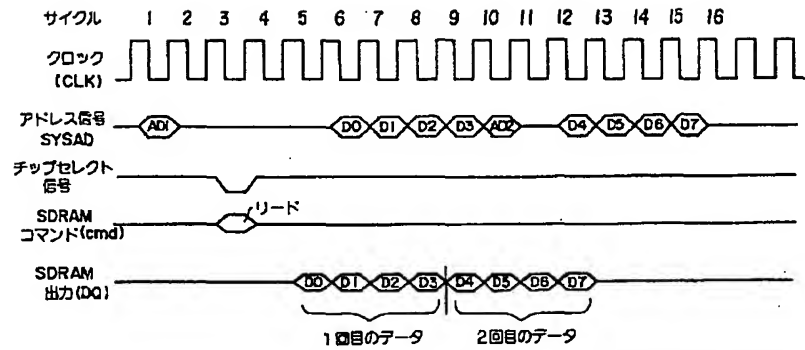
【図2】RAS及びCASの動作タイミングを示すタイミングチャートである。

【符号の説明】

D0~D7…データ

AD1, AD2…アドレス

【図1】



【図2】

